The CDISC Laboratory Data Interchange Standard: Using SAS® To Read and Write CDISC Compliant Data Sets and Files

Margaret Hung, GlaxoSmithKline
Philip M. Pochon, Covance Central Laboratory Services, Inc.

ABSTRACT

The Clinical Data Interchange Standards Consortium (CDISC) has released for public review and comment its data model for clinical trial laboratory data interchange. The version 1 release of the laboratory data interchange model defines standards for SAS® data sets and ASCII files. This paper first reviews the key features of the CDISC laboratory data interchange model and discusses planned future extensions to the model. It then focuses on the experiences gained during early testing of the model. This testing evaluated processes for transferring central laboratory data to pharmaceutical companies. The testing used base SAS® to generate and read SAS® data sets and ASCII files. Testing was performed across multiple versions of SAS®, and multiple operating systems. Testing methods are reviewed and sample SAS® code is examined during a discussion of the issues discovered during testing.

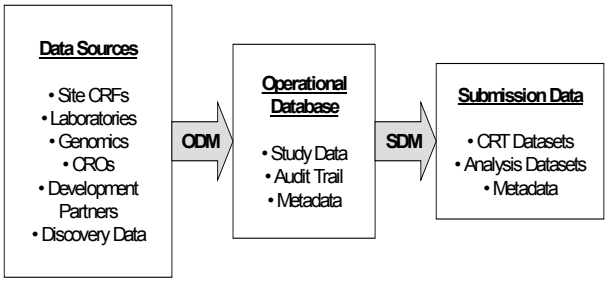
INTRODUCTION

Formed in 1997, CDISC is an open, multidisciplinary, non-profit organization committed to the development of industry standards to support the electronic acquisition, exchange, submission and archiving of clinical trials data and metadata for medical and biopharmaceutical product development.

The mission of CDISC is to lead the development of global, vendor-neutral, platform-independent standards to improve data quality and accelerate product development in the pharmaceutical industry.

CDISC aims to achieve its mission through the development of standard data models designed to support the end-to-end data flow of clinical trials from the sources of the data into an operational database and through to analysis and submission, as shown below (Figure 1).

Figure 1



ODM = Operational Data Modeling
SDM = Submissions Data Modeling

Formed in 2000, the Laboratory Data Team (LAB) has as its mission the development of a standard model for the acquisition and interchange of clinical trials laboratory data.

The members of the LAB team represent a broad cross-section of stakeholders from within the biopharmaceutical industry who have an interest in clinical laboratory data. The team is currently comprised of representatives from 4 pharmaceutical companies, one biotechnology company, one CRO and 3 central laboratories. The membership of the team incorporates expertise from a variety of clinical, technical and statistical disciplines and a variety of different perspectives including academic, commercial and European.

It is estimated that the cost to the industry per year simply of laboratory data interchange itself is at least $150m and that between approximately 30% and 60% of that cost could be saved from the use of a standard. However it is recognized that the real – and substantially greater - value that faster and higher quality data interchange brings is in terms of time savings in an industry with running costs estimated at $1m per day.

The CDISC LAB team has developed a standard model for laboratory data interchange. In December of 2001, this model was posted to the CDISC Web site and made available for public comment.

CDISC LABORATORY DATA INTERCHANGE MODEL

The CDISC laboratory data Content Model has a main core designed to handle 'simple' laboratory data with the classic 'one test, one result' data structure. Extensions will be added to the code model to handle more 'complex' laboratory data such as Microbiology and Pharmacogenomics.

The core model is separated into 10 logical levels as follows:

- Good Transmission Practice
- Study
- Site
- Subject
- Visit
- Accession
- Container
- Panel
- Test
- Result

These levels were chosen because they follow the recognizable hierarchy of clinical laboratory data. The hierarchical nature of the model levels will significantly reduce data redundancies when hierarchical implementations of the model are finalized.

Within each level, required and/or optional fields contain data appropriate to that level. Within the core model, 15 fields are considered required, while all other fields are optional.

Extending the model for more complex data structures involves:

- adding fields to the current model levels
- adding levels with new fields

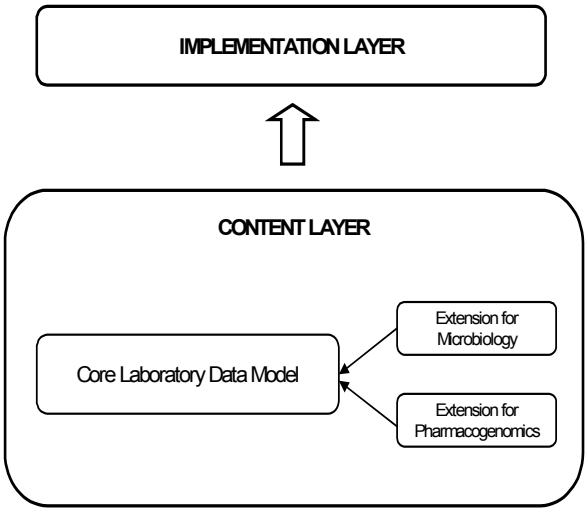
Since the purpose of the model is to improve the interchange of

laboratory data it is important to consider not only the structure of the model and its fields, but also the population of those fields. Accordingly, for some fields, code lists have been suggested. The purpose of these code lists is to offer a higher degree of standardization and so further improve the reliability and accuracy of data interchange.

A 'multi-layer' model was then developed whereby the first layer would be the content layer and above that would be an implementation layer, the idea being that the content would not change but the implementation could (See Figure 2).

The advantage of this approach is that it offers flexibility but retains control: it doesn't make the use of the model dependent upon any one implementation and if different implementations are used the content remains the same so the standard still applies.

Figure 2



The initial release of the core model is as a flat file format which can be implemented as a bar delimited ASCII file or as a SAS® transport data set.

Given the hierarchical nature of the model, a more natural implementation is as a hierarchical XML file. This implementation method is being developed and it is expected to be released for public comment in 2002.

CURRENT SITUATION AT GSK & COVANCE

GLAXOSMITHKLINE (GSK)

GSK is a research-based pharmaceutical and healthcare company. It is a result of a merger in 2001 of GlaxoWellcome (GW) and SmithKline Beecham (SKB). There are four major UK/US sites with various worldwide operating companies

GW and SKB each uses distinctly different dB systems, software tools, standards, processes, and SOP's. The merger of the two companies present a challenge to develop viable systems, workable standards, and feasible processes for capturing, managing, and reporting clinical trial data for the new company. GW data are stored and managed in a proprietary DBMS called GOLD whereas SKB uses Clintrial (CT4 & CT3) as its DBMS

Lab data are managed in multiple formats in GOLD and CT3, CT4 with completely different processes. All require up-front manipulation prior to upload. Data can be in ASCII of CSV or fixed format, SAS®, and Excel. A new way of processing and managing lab data is in progress at the time this paper was being written.

COVANCE CLS

Covance Central Laboratory Services (Covance CLS) is a global provider of clinical trial laboratory services for the pharmaceutical industry. Five labs worldwide provide standard laboratory services, with specialty labs at the two largest sites (Indianapolis and Geneva) providing microbiology, flow cytometry, genomic, pharmacogenomic and pharmacokinetic capabilities. A sixth lab specializes in ECG data.

Each laboratory maintains its own resulting database, with most standard (the classic 'one test, one result' data structure) results now being shadowed real-time into a central repository.

Covance CLS maintains 400-500 active file formats at any one time for data transmittal to client pharmaceutical companies. Data transmittals are in ASCII flat file format or SAS® transport data sets. Data transmittal to, and reception from referral labs further complicates the issue.

CDISC LAB DATA TESTING PROCESS

Prior to posting the laboratory data model for public comment, the model was tested internally by the members of the LAB team. Each biotech and central laboratory paired with one or more pharmaceutical companies and conducted parallel testing using active study data.

THE COVANCE EXPERIENCE – DATA SOURCE AND SENDER

Covance CLS transmitted test data to three pharmaceutical companies (including GSK) using SAS® Version 8.1 to create:

- A SAS® transport data set with cumulative study data for blood chemistry, urine chemistry and hematology
- An ASCII flat file with cumulative study data for blood chemistry urine chemistry, serology and hematology. This test included timed blood draws.
- An ASCII flat file with cumulative study data for blood chemistry, serum electrolytes, hematology and one human genotype/phenotype test.

General information from tests performed by other members of the CDISC LAB team is included in this paper, but the focus is upon the testing actually performed by the authors.

The general testing methodology was that a data transmission file was created for a study using that study's active format. The same data was then extracted into a standard format ASCII file which was read into SAS®, which then processed the data and produced a SAS® transport data set or ASCII file in CDISC format. The ability to replicate the study specific formatted data in CDISC format was evaluated, and the processes involved in creating and reading the CDISC files documented and critiqued.

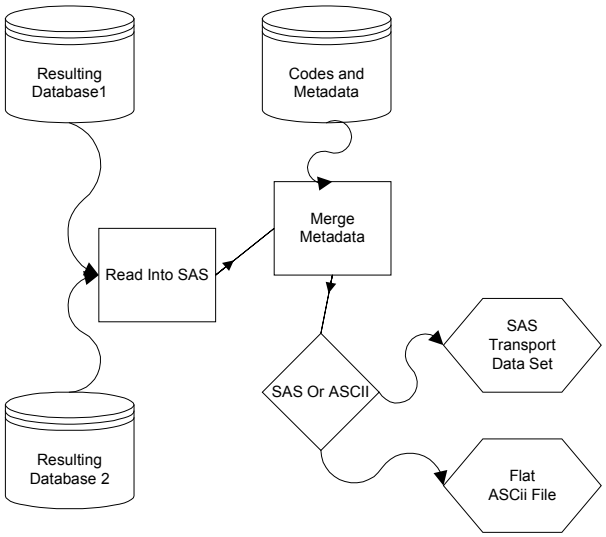
CREATING A CDISC COMPLIANT DATA SET

The initial release of the CDISC Laboratory data model is intended to cover standard "one test, one result" laboratory data. The process of creating a CDISC compliant data transfer file is one of converting individual corporate data structures into the CDISC structure. In most cases, this involves three major activities:

1. Creating the Good Transmission Practice (GTP) information about the transmission
2. Separating data that may be aggregated within the corporate data structure into its CDISC components
3. Translating corporate values into CDISC values or codes.

The second and third activities are the most critical. In setting up the testing process, Covance chose to start from an assumption that data disaggregation and translation must be metadata driven, rather than hard coded. This allows additions and changes to be made at the data, not code level, and limits version control to just a few metadata tables. The general process is shown in Figure 3.

Figure 3



Two issues that occurred during SAS®-to-SAS® testing impacted several fields within the CDISC model.

One issue was the CDISC date format. All dates in the CDSIC model are to be formatted as 'YYYY-MM-DD'. In SAS® version 8, the 4 digit year and the dashes can be obtained using the format 'YYMMDD10.' When SAS® Version 6.12 attempted to read this date format it generated an error message. The closest SAS® version 6 format is 'YYMMDD8.', which does produce the dashes, but only a two digit year.

The second issue was the length of SAS® character fields. When creating .XPT files with PROC COPY, the length of character fields is limited to the SAS® version 6 maximum of 200 characters. Several CDSIC character fields are set to 2048 characters. In all SAS® transport file tests, these fields were set to 200 characters.

Good Transmission Practice (GTP) Level

The GTP level of the CDISC Lab Data model presented few problems during testing.

One general issue is that the three identifier fields (Transmission source in the GTP Level, Central Laboratory in the Accession Level) and Performing Laboratory in the Test level). Require ID values and allow names. The is no standard for such ID values, so in both cases, the ID and name were set to the same value in testing.

At the time of testing, the Transaction Type was part of the GTP level. The Transaction Type indicates how the record should be processed when it is imported into the recipient database. The possible values for Transaction Type are:

- D – Delete existing record
- I – Insert new record
- R – Retransmit existing record without changes
- U – Update existing record

To simplify testing and to simulate the most common transmission practice, cumulative data transmission files were created with all records being set to 'I' (Insert).

Within a flat file structure, incremental transmissions can set the transaction type ('D', 'I' or 'U') for each record. Incremental extracts from the Covance data bases set this flag on each record, so flat file structure will have few problems. However, once a hierarchical structure is in use, the location of this field in the highest level of the model would necessitate creating a separate file for each transaction type. As a result of this issue, the Transaction Type was provisionally moved to the bottom level (Result) of the model. This allows result values to be properly handled, but creates problems when intermediate level information (e.g. Patient Initials or Collection Date) must be updated.

Study Level

The study ID and Name were transmitted directly from the Covance CLS database values.

Site Level

The investigator ID and name were transmitted directly from the Covance CLS database values.

The site ID was not used in testing, but would be available for some studies through recursive parenting of investigators. This is one case where processes we currently consider optional in the protocol definition may need to become standard in order to properly meet CDISC model needs.

Subject Level

The Subject ID, Subject Initials, Subject Sex and Subject Date of Birth were transmitted directly from the Covance CLS database values. One test used Screen ID, and this was also transmitted directly from the database values.
One test study did collect Subject Race. These values were transmitted directly from the database values. In future work, a metadata table will need to match Covance Ethnicity codes to the HL7 Race Vocabulary values.

Visit Level

The Visit Number, Visit Name and Visit Type were transmitted directly from the Covance CLS database values. The Subject Age at Visit was computed from the Subject Date of Birth and the Visit Collection Date using the following SAS® code (See Figure 4), where LBACTDT is the Actual Collection Date from the Container Level (see below):

Figure 4

```
DATA WORK.CDISC;
   ATTRIB SUBJAGE LENGTH=8;
   SET WORK.CDISC;

SUBJAGE=INT(INTCK('MONTH',SUBJDOB,LBACTDT)/
        12);
      IF MONTH(SUBJDOB)=MONTH(LBACTDT) THEN
         SUBJAGE=SUBJAGE-(DAY(SUBJDOB) >
```

```
            DAY(LBACTDT));
RUN;
```

Accession Level

As discussed under the GTP level, the Central Laboratory ID and Name were set to the same value. The Accession Number, Last Active Date and Last Active Time were transmitted directly from the Covance CLS database values.

Container Level

The Container ID, Actual Collection Date, Actual Collection Time, Received Date and Received Time and Specimen Condition were transmitted directly from the Covance CLS database values.

The Specimen Material, although an optional field within the model, was felt to be an important data element that should be transmitted. The current Covance CLS data model combines a number of CDISC data elements into a single entity, the test code. An extreme example would be a glucose draw for a diabetes study: "Blood Glucose, Fasted, 2 hour". Sample material, test, fasting status and planned collection time are all encapsulated within the test code text.

In order to separate the data elements combined in test codes, a metadata table was developed for all test codes included in the study protocols used in testing. Each test code was decomposed so that each CDISC element could be stored in a separate column. The above diabetes example is shown as row 2 in Figure 5, which displayed a set of four timed draws (Baseline, two hour, four hour and eight hour):

Figure 5

TSTNAM	SPCMMNAM	FASTSTTS	PCOLLINT	TSTDTTYP
Glucose	Blood	Yes		Numeric
Glucose	Blood	Yes	2 hr	Numeric
Glucose	Blood	No	4 hr	Numeric
Glucose	Blood	No	8hr	Numeric

The planned Collection Time (PCOLLTM) had to be computed using the normative interval expressed as minutes, hours or days. The test data type (TSTDTTYP) is a normative data flag used to determine the data type of the result (see below: Result Level).

This metadata table was merged into the data read from the database, merging by the Covance CLS test code. During testing, no attempt was made to use LOINC (Logical Observation Identifier Names and Codes) or HL7 codes, the Covance code was transmitted for ID value. These codes will need to become part of this metadata merge process.

Panel Level

The Panel ID and Panel Name were transmitted directly from the Covance CLS database values.

Test Level

The Performing Lab Name, Lab Test Name, Test Name, Test Status, testing Date and Testing Time were transmitted directly from the Covance CLS database values. The Test Status was translated Covance CLS 'C' = Cancelled to CDISC 'X' = Cancelled).

The Performing Lab ID, Lab Test ID and Test ID were transmitted as the Covance CLS data values, not LOINC codes.

In one test study four non-study tests were performed, and the Test Type field was translated from the internal Covance CLS codes.

The Test Level Comments field must be used for both cancelled test comments (in a hierarchical data model, a cancelled test will have no result level records) and for result level comments (in the current CDSIC standard model, which assumes one test – one result, there is no result comment field).

Result Level

The CDSIC Result Level is broken into four blocks: Original Result, Conventional Result, SI Result and Reported Result. Data Issues. In each block the Units of Measure, a Numeric Result or Text Result, and the High and Low Reference Range Values are to be transmitted. A final suite of fields then covers status, flags and reported date and time.

For each block, the Units of Measure, and the High and Low Reference Range Values (when appropriate) were transmitted directly from the Covance CLS database values.

For each block, the result value was initially read into a character field. The test code metadata for that result was merged in to provide three lookup fields. The first field was the standard data type ("Numeric", "Text", "Code" or "Range"). If the standard data type was ", "Text", "Code" or "Range", then the CDISC Original Result Type field was set accordingly, and the value was placed into the text result field for that block.

If the standard data type was "Numeric", then four checks were performed on the original result:

1. If the original result contained any letter value, the CDISC Original Result Type field was set to "T" (Text) and the value was placed into the text result field for that block.
2. If the test could be bounded, a metadata field defined the normal low boundary value. If the original result was equal to this value, or contained a "<" sign, the CDISC Original Result Type field was set to "L" (Low Boundary) and the value was placed into the text result field for that block.
3. If the test could be bounded, a metadata field defined the normal high boundary value. If the original result was equal to this value, or contained a ">" sign, the CDISC Original Result Type field was set to "H" (High Boundary) and the value was placed into the text result field for that block.
4. If the original result contained a "-", the CDISC Original Result Type field was set to "R" (Range) and the value was placed into the text result field for that block.
5. Otherwise the value was a true numeric, the CDISC Original Result Type field was set to "N" (Numeric) and the value was placed into the numeric result field for that block.

The code for this checking process controlled original, conventional, SI and reported results. An example checking just the original results is shown in figure 6.

Figure 6

```
SELECT (UPCASE(RSLT_TYP));
  WHEN ('CODE') DO;
    ORGRESC=COV_RSLT;
    ORGRTYPE='C';
  END;
  WHEN ('TEXT') DO;
    ORGRESC=COV_RSLT;
    ORGRTYPE='T';
  END;
  WHEN ('RANGE') DO;
    ORGRESC=COV_RSLT;
    ORGRTYPE='R';
```

```
    END;
    WHEN ('NUMERIC') DO;
      IF (0^=INDEXC(COV_RSLT,
        'ABCDEFGHIJKLMNOPQRSTUVWXYZ'))
        THEN DO;
        ORGRESC=COV_RSLT;
        ORGRTYPE='C';
      END;
      ELSE IF (0^=INDEXC(COV_RSLT,'<')) OR
        (COV_RSLT=LOW_BND) THEN DO;
        ORGRESC=CVN_RSLT;
        ORGRTYPE='L';
      END;
      ELSE IF (0^=INDEXC(COV_RSLT,'>')) OR
        (COV_RSLT=HIGH_BND) THEN DO;
        ORGRESC=COV_RSLT;
        ORGRTYPE='H';
      END;
      ELSE IF (0^=INDEXC(COV_RSLT,'-')) THEN DO;
        ORGRESC=COV_RSLT;
        ORGRTYPE='R';
      END;
      ELSE DO;
        ORGRESN=COV_RSLT;
        ORGRTYPE='N';
      END;
    END;
    OTHERWISE;
  END;
```

The Alert Flag, Delta Flag, Exclusion Flag, reported Date and Reported Time were transmitted directly from the Covance CLS database values.

WRITING CDISC COMPLIANT OUTPUT

Preparation of the data into a CDISC compliant flat data record was a single processing stream. Only once the record had been created was the question of the file type addressed. For tests that used SAS® Transport data sets, PROC COPY was used to create a SAS® Transport data set. The Transport data set mode was used because the GSK test used different versions of SAS® (8.1 to create the data and 6.12 to read it) and different operating systems (Windows NT 4.0 to create the data and Unix to read it).

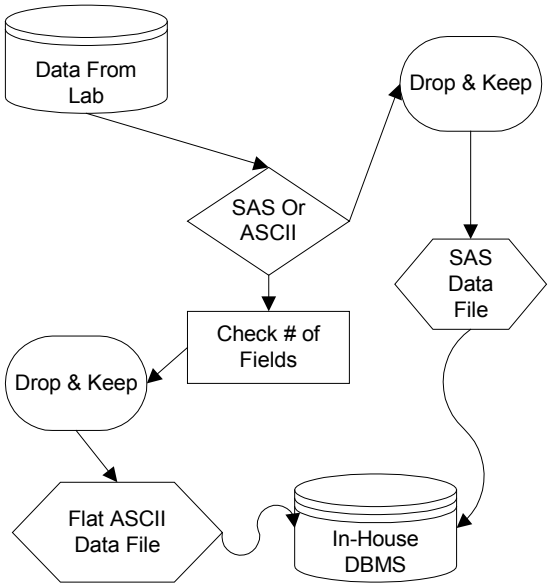
The two tests that used the flat ASCII file format used a DATA step with PUT statements to write bar delimited records to the file.

THE GSK EXPERIENCE – THE SPONSOR AND RECEIPIENT

GSK performed the testing with two central core labs – Covance and Q Lab. The former sent GSK lab data in SAS® datasets whereas the latter sent data in ASCII bar delimiter format. Partial production data were used as test data. The test data were compliant to the study specifications and guidelines. Both data files were loaded into GSK/GW GOLD system using GW processes. All the transformation was done via SAS® programs in SAS® 6.12.

GSK, being the sponsor, has one major advantage over a lab vendor. It was required to do minimum transformation on the incoming data. Many CDISC fields including numerous mandatory fields can be ignored. GSK only picks the fields that are required by its in-house DBMS and specified by the protocol. Furthermore, for GSK/GW, it is a convention that minimum checking is to be performed on external electronic core lab data. Figure 7 shows the process flow.

Figure 7



READING A CDISC COMPLIANT DATA SET

For the ASCII delimiter file, a program first checked that all the fields are included which is basically counting the number of "bars" present in the file. This is to ensure that the correct fields will be read-in later in the transformation. For the SAS® dataset, this step is not necessary.

After the initial checking, another program was used to read in only the fields that GSK needed. The GW GOLD system needed only fifteen (15) fields from the lab data model. The rest of the fields were dropped.

These 15 fields were: Study ID (SITEID), Investigator ID (INVID), Subject ID (SUBJID), Visit ID (VISITNUM), Accession ID (ACCSNNUM), Panel ID (PANELNUM), Test ID (LTSTNUM), Testing Date (TSTDT), Testing Time (TSTTM), Conventional Units (CNVUNIT), Conventional Numeric Result (CNVRESN), Conventional Text Result (CNVRESC), Conventional Normal Reference Range Low (CONVNRLO), Conventional Normal Reference Range High (CNVNRHI), Alert Flag (ALRTFLAG)

CONVERTING A CDISC COMPLIANT DATA SET FOR IN-HOUSE USE

After the "dropping and retaining", another routine was set up to transform the data compatible to the in-house GW system. Required keys were plugged in according to the study visits, dates were reformatted to DDMMMYY, and time fields changed to time5.

GSK did not use the CDISC suggested code lists. The codes were not checked prior to loading but were checked internally once the data was loaded.

In addition, the CDISC SAS® variable names were changed to GW in-house fieldnames, e.g., for Subject ID, CDISC's SUBJID was changed to GW's SUBJECT, VISITNUM to SESS, etc.

The following is a sample code outputting an ASCII file.

Figure 8

```
put @1 ptid @9 invid @ ;
      %writeout(14,subject,7.);
      %writeout(21,repeat,2.);
      %writeout(23,sess,4.1);
      put @27 specid @;
      %writeout(43,vsdt,date7.);
      %writeout(50,sltm, time5.);
      ……..
      put;
```

VALIDATING THE END RESULT

Both the ASCII and SAS data files were checked by standard in-house lab utilities and were successfully loaded into the system. Random sample data was checked against the original data file. No discrepancy was found.

LESSIONS LEARNED

COVANCE CLS

One of the significant lessons learned during testing at Covance CLS was that in order for standard data formats to work, certain standards we now follow will need to be revisited and perhaps revised to bring our data into conformity with the standard transmittal model. In addition, certain standards we now consider optional in study design will need to be implemented for all studies.

The process of translating Covance CLS data structures into CDISC compliant data structures revealed a number of points where our data model requires translation and disaggregation. The ability to perform these conversions was simplified by the use of metadata lookup tables even within the limited scope of the initial testing. The cost and effort to build and maintain standard tables for the codes, translations and disaggregation processes needed for all Covance CLS data will be high, but this must be compared to the current costs of building and maintaining hundreds of study specific data formats and programs.

The need for clearly designed data specifications was apparent at Covance CLS as we moved from test to test with three different pharmaceutical companies. Although none of the required or "major" fields in the model were dramatically different from test to test, the availability of data for several of the optional fields varied from test to test. This issue will become even more important as we move into tests that merge referral lab data with central lab data.

GSK

For GSK, the experience of working with the CDISC lab model was painless and rewarding. Since GSK already has established standard formats for lab data, the process of reading in data in CDISC format and writing out the GSK compliant data file was rather straightforward. The policy of performing minimum data validation on external electronic lab data further simplified the testing.

One of the significant lessons learned during testing was that in order for standard data formats to work, certain standards we now consider as optional in study design will need to be implemented for all studies. For example, one standard format for capturing date and time and one convention for visits would reduce time in data transformation. With CDISC lab data model in place, one standard tool can then be developed to read and output data more efficiently.

Clearly defined up-front specifications from the sponsor to the vendor also help in reducing miscommunication. For example, testers from both GSK and the labs did not clearly specified what should be in the Test ID field. As a result, extra code had to be written to translate the test codes to GSK's own test names.

On-going two-way communication between the data manager from the lab and GSK further clarifies data details and helps streamline the data exchange process.

Lastly, in order to accept and adopt the CDISC lab data model, resources have to be allocated to create and implement company-wide standards, develop generic applications in transforming and accepting lab data, and train appropriate personnel to manage the data.

IMPLEMENTATION

COVANCE

Covance maintains a strong commitment to CDSIC data standards in general, and at Covance CLS to the Laboratory Data Model. Through active participation in the CDSIC Lab committee we expect to further refine the base data model, assist in developing extensions to the base model, and to implement CDISC compliant data transmission formats as these become production approved.

GSK

GSK has reviewed the CDISC lab model and decided to adopt the standards in progressive phases. At the time when this paper was written, GSK has agreed to adopt the following:

- CDISC SAS® variable names
- Lab result data in numeric and character fields
- Fasting indicator included as a separate item
- "Planned Collection Text" used as part of a Time ID
- Reference range data included in lab data file

Future consideration includes CDISC suggested code lists and LOINC names and codes.

CONCLUSION

Participants of the tests reported that the model is very easy to use because the structure of the different levels of laboratory data within it is clear and logical.

As general conclusions from all testing of the base model, from the central laboratory perspective this made the population of the data fields straightforward and unambiguous because there were clear relationships between what the model requires and where the data resides in clinical data management systems. The ease of use of the model both in terms of understanding what fields are required and how those fields should be populated made the implementation of programs to create data files a very straightforward process with no requirements for complex programming. Most notably this made significant savings in terms of time and resulted in very fast implementations with no sacrifice in quality.

The pharmaceutical companies reported that the data was easily extracted from the files and that the content of the model was quite adequate for their requirements. It was noted that the logical organization of the data and the ease of its extraction would certainly allow very straightforward translations into existing technical infrastructures and applications and that the advantage of a single standard to streamline the acquisition and processing of data was clear.

REFERENCES

CDSIC Lab Committee, 2001 “CDISC Laboratory Data Interchange Standard” Available at http://www.cdisc.org/

Kush, Rebecca Daniels 2001 “The Cost of Clinical Data Interchange in Clinical Trials: A CDSIC White Paper” Available at http://www.cdisc.org/

ACKNOWLEDGMENTS

The authors wish to acknowledge the support and assistance of the following CDISC LAB team members who assisted in testing. The views expressed in this paper remain those of the authors.

Julian Banks, formerly with Q Laboratory, Quintiles
Scott Getzin, Eli Lilly and Co., Inc.
John Harkins and Philip Stipcevich, Merck & Co., Inc.

TRADEMARK

CONTACT INFORMATION

Margaret Hung
GlaxoSmithKline
Five Moore Drive
RTP, NC 27709
Mh34102@gsk.com

Philip M. Pochon
Covance CLS
8211 SciCor Drive
Indianapolis, IN 46214
phil.pochon@covance.com